

Ộ NÔNG NGHIỆP VÀ MÔI TRƯỜNG TRƯỜNG
ĐẠI HỌC THỦY LỢI



BÀI TẬP LỚN

PHÁT TRIỂN ỨNG DỤNG CHO THIẾT BỊ DI ĐỘNG

**ĐỀ TÀI: ỨNG DỤNG GỌI MÓN CHO QUÁN ĂN
CN FOOD**

Giáo viên hướng dẫn: ThS. Kiều Tuấn Dũng

Sinh viên thực hiện:

STT	Mã sinh viên	Họ và tên	Lớp
1	2251061783	Trịnh Hải Hoàn	64CNTT2
2	2251061878	Nguyễn Quang Thái	64CNTT2

Hà Nội,

năm 2025

**BỘ NÔNG NGHIỆP VÀ MÔI TRƯỜNG
TRƯỜNG ĐẠI HỌC THỦY LỢI**



BÀI TẬP LỚN

PHÁT TRIỂN ỨNG DỤNG CHO THIẾT BỊ DI ĐỘNG

ĐỀ TÀI: ỨNG DỤNG GỌI MÓN CHO QUÁN ĂN CN FOOD

STT	Mã Sinh Viên	Họ và Tên	Ngày Sinh	Điểm	
				Bảng Số	Bảng Chữ
1	2251061783	Trịnh Hải Hoàn	28/07/2004		

CÁN BỘ CHẤM THI

Hà Nội,

năm 2025

LỜI NÓI ĐẦU

Trong thời đại công nghệ phát triển nhanh chóng, việc ứng dụng các giải pháp phần mềm vào hoạt động kinh doanh, dù là ở quy mô nhỏ, đang dần trở nên phổ biến. Đặc biệt trong lĩnh vực ăn uống, những ứng dụng gọi món đơn giản đã và đang hỗ trợ hiệu quả cho các quán ăn nhỏ, hộ kinh doanh cá thể hay các mô hình khởi nghiệp sinh viên trong việc phục vụ khách hàng, quản lý đơn hàng, và theo dõi doanh thu.

Đề tài "Ứng dụng gọi món cho quán ăn CN Food" được xây dựng nhằm mục đích tạo ra một hệ thống đơn giản, thân thiện, dễ sử dụng cho cả người dùng và người quản lý. Ứng dụng cho phép người dùng có thể đăng ký, đăng nhập, xem danh sách món ăn, tìm kiếm món ăn, thêm vào giỏ hàng và xác nhận đơn hàng. Đồng thời, người quản lý cũng có thể dễ dàng thêm, sửa, xóa các món ăn trong thực đơn và xem thống kê doanh thu theo từng tháng, bao gồm cả top 3 món ăn bán chạy nhất.

Ứng dụng được phát triển trên nền tảng Android với ngôn ngữ lập trình Java, sử dụng Firebase Realtime Database để lưu trữ dữ liệu và Firebase Authentication để xử lý đăng nhập người dùng. Với việc áp dụng các công nghệ phổ biến và phù hợp cho các dự án học tập, nhóm mong muốn không chỉ rèn luyện kỹ năng lập trình ứng dụng di động, mà còn xây dựng được một sản phẩm thực tế có thể sử dụng trong môi trường kinh doanh nhỏ lẻ.

Mặc dù ứng dụng còn nhiều điểm hạn chế so với các phần mềm thương mại chuyên nghiệp, nhưng đề tài này đã giúp nhóm tiếp cận quy trình phát triển ứng dụng từ thiết kế, xây dựng đến triển khai và thử nghiệm thực tế. Đây sẽ là bước đệm quan trọng để nhóm tiếp tục hoàn thiện và mở rộng các tính năng trong tương lai.

MỤC LỤC

PHÁT TRIỂN ỨNG DỤNG CHO THIẾT BỊ DI ĐỘNG ĐỀ TÀI: ỨNG DỤNG.....	2
LỜI NÓI ĐẦU.....	3
Chương 1. TỔNG QUAN VỀ ĐỀ TÀI.....	5
1.1 Giới thiệu về đề tài.....	5
1.2 Mục tiêu của đề tài.....	5
1.3 Phạm vi của đề tài.....	5
1.4 Phân chia nhiệm vụ.....	6
Chương 2. KIẾN TRÚC VÀ CÔNG NGHỆ.....	7
2.1 Kiến trúc hệ thống.....	7
2.2 Giới thiệu về Công nghệ phát triển.....	7
Chương 3. XÂY DỰNG ỨNG DỤNG.....	8
3.1 Thiết kế Figma.....	8
3.2 Thiết kế CSDL.....	13
3.2. Giao diện ứng dụng.....	14
3.4 Code minh họa các chức năng cốt lõi.....	17
KẾT LUẬN.....	18
1. Kết quả đạt được.....	18
2. Nhược điểm.....	18

Chương 1. TỔNG QUAN VỀ ĐỀ TÀI

1.1 Giới thiệu về đề tài

Trong bối cảnh công nghệ ngày càng phát triển, nhu cầu sử dụng các ứng dụng di động để hỗ trợ hoạt động kinh doanh và phục vụ khách hàng ngày càng trở nên phổ biến. Đặc biệt trong lĩnh vực ăn uống, việc xây dựng các ứng dụng đặt món giúp tối ưu hóa quy trình phục vụ, nâng cao trải nghiệm khách hàng và quản lý hiệu quả hoạt động của quán ăn.

Đề tài “Ứng dụng gọi món cho quán ăn CN Food” được xây dựng với mục đích hỗ trợ một quán ăn nhỏ trong việc quản lý menu, nhận đơn hàng, thống kê doanh thu và cải thiện tính chuyên nghiệp trong vận hành. Ứng dụng hoạt động trên nền tảng Android, cho phép khách hàng xem menu, đặt món và xác nhận đơn hàng một cách dễ dàng. Dữ liệu của hệ thống được quản lý và đồng bộ theo thời gian thực thông qua nền tảng Firebase của Google.

Thông qua đề tài này, nhóm mong muốn ứng dụng kiến thức đã học vào thực tiễn, đồng thời rèn luyện kỹ năng phát triển ứng dụng Android, làm việc với cơ sở dữ liệu thời gian thực và thiết kế giao diện người dùng thân thiện.

1.2 Mục tiêu của đề tài

Đề tài hướng tới mục tiêu xây dựng một ứng dụng di động đơn giản, tiện lợi, phù hợp với mô hình hoạt động của quán ăn nhỏ. Cụ thể, ứng dụng cần đạt được các mục tiêu sau:

- Xây dựng giao diện trực quan, thân thiện với người dùng, phù hợp với cả người sử dụng và người quản lý quán ăn.
- Cho phép người dùng xem thực đơn, thêm món vào giỏ hàng, xác nhận gọi món một cách nhanh chóng và thuận tiện.
- Hỗ trợ người quản lý thực hiện các thao tác thêm, sửa, xóa món ăn trong menu một cách dễ dàng.
- Tích hợp tính năng đăng ký và đăng nhập người dùng thông qua Firebase Authentication nhằm đảm bảo tính bảo mật và cá nhân hóa trải nghiệm người dùng.
- Lưu trữ và đồng bộ dữ liệu theo thời gian thực thông qua Firebase Realtime Database.

- Thống kê đơn hàng và doanh thu theo tháng, đồng thời xác định top 3 món ăn được gọi nhiều nhất để hỗ trợ công tác quản lý.

1.3 Phạm vi của đề tài

Đề tài tập trung phát triển một ứng dụng gọi món đơn giản dành cho các quán ăn quy mô nhỏ, trong đó phạm vi cụ thể bao gồm:

- Ứng dụng được xây dựng trên nền tảng Android, sử dụng ngôn ngữ lập trình Java.
- Sử dụng Firebase Authentication để quản lý đăng ký, đăng nhập người dùng.
- Dữ liệu món ăn, đơn hàng, giỏ hàng và thống kê được lưu trữ trên Firebase Realtime Database.
- Ứng dụng cho phép:
 - Xem danh sách món ăn.
 - Tìm kiếm món ăn theo tên.
 - Thêm món vào giỏ hàng, chỉnh sửa số lượng và xác nhận đặt món.
 - Quản lý món ăn: thêm, sửa, xóa món.
 - Xem thống kê doanh thu và top 3 món ăn bán chạy theo tháng.
- Không triển khai tính năng thanh toán online, phân quyền người dùng nâng cao hay quản lý kho nguyên liệu.
- Ứng dụng chưa tối ưu hóa cho hoạt động trên nhiều loại thiết bị và chưa có phần quản lý đặt bàn hoặc giao hàng.

1.4 Phân chia nhiệm vụ

<<Bảng phân chia nhiệm vụ>>

STT	Họ và tên	Mã SV	Nhiệm vụ chính
-----	-----------	-------	----------------

1	Trịnh Hải Hoàn	2251061783	<ul style="list-style-type: none"> - Thiết kế figma các màn hình chính của ứng dụng. - Thiết kế các item giao diện XML. - Xây dựng chức năng đăng ký, đăng nhập với Firebase Authentication. - Xây dựng chức năng giỏ hàng, tăng giảm số lượng món ăn. - Tạo navigation menu chuyển đến các Activity như chỉnh sửa món ăn, thống kê, đăng xuất. - Xây dựng Activity chỉnh sửa món ăn: tìm kiếm theo tên, xóa món, sửa món. - Viết và hoàn thiện báo cáo.
2	Nguyễn Quang Thái	2251061878	<ul style="list-style-type: none"> - Góp ý giao diện và trải nghiệm người dùng. - Kết nối và xử lý dữ liệu với Firebase Realtime Database & Authentication. - Xây dựng Activity thống kê. - Hỗ trợ xử lý logic nghiệp vụ. - Kiểm tra, thử nghiệm và tìm lỗi trong giao diện, tính năng ứng dụng.

Chương 2. KIẾN TRÚC VÀ CÔNG NGHỆ

2.1 Kiến trúc hệ thống

Ứng dụng gọi món CN Food được xây dựng theo mô hình kiến trúc Client – Server:

- Client là ứng dụng di động Android cài đặt trên thiết bị của người dùng.
 - Giao diện người dùng được xây dựng bằng XML và Java.
 - Cho phép người dùng đăng ký, đăng nhập, xem danh sách món ăn, thêm vào giỏ hàng, gửi đơn và xem thống kê.
 - Có chức năng tìm kiếm món ăn, chỉnh sửa thực đơn (nếu là quản lý).
- Server là nền tảng Firebase Realtime Database và Firebase Authentication đóng vai trò lưu trữ dữ liệu và xử lý xác thực người dùng. Firebase cung cấp khả năng đồng bộ dữ liệu theo thời gian thực, giúp ứng dụng luôn hiển thị thông tin mới nhất.

2.2 Giới thiệu về Công nghệ phát triển

Trong quá trình xây dựng ứng dụng CN Food, nhóm đã sử dụng các công nghệ và công cụ sau:

- Ngôn ngữ lập trình: Java – được sử dụng chính để xây dựng logic ứng dụng trên nền tảng Android.
- Android Studio: Công cụ phát triển chính, hỗ trợ thiết kế giao diện (XML), viết mã Java, kiểm thử và debug ứng dụng.
- Firebase Authentication: Cung cấp dịch vụ xác thực người dùng (đăng ký, đăng nhập) nhanh chóng, dễ tích hợp.
- Firebase Realtime Database: Lưu trữ và đồng bộ dữ liệu thời gian thực giữa người dùng và hệ thống (menu, đơn hàng, thống kê...).
- Thư viện hỗ trợ:
- Glide: Dùng để load và hiển thị hình ảnh món ăn.
- RecyclerView, CardView: Hiển thị danh sách món ăn và các thông tin động theo dạng danh sách cuộn mượt mà.

Chương 3. XÂY DỰNG ỨNG DỤNG

3.1 Thiết kế Figma

<<Link Gitub và ảnh kết xuất Figma các màn hình>>

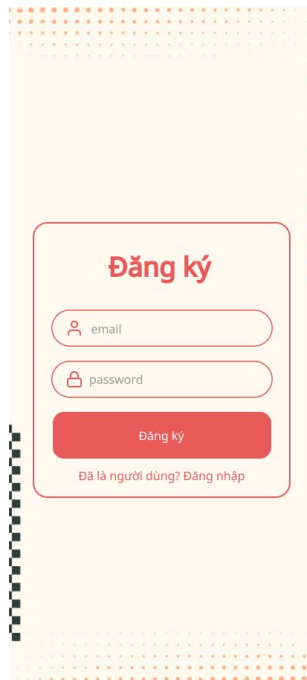
Link Github: [haihoan2874/BTL_Mobile](https://github.com/haihoan2874/BTL_Mobile)

Link Figma:

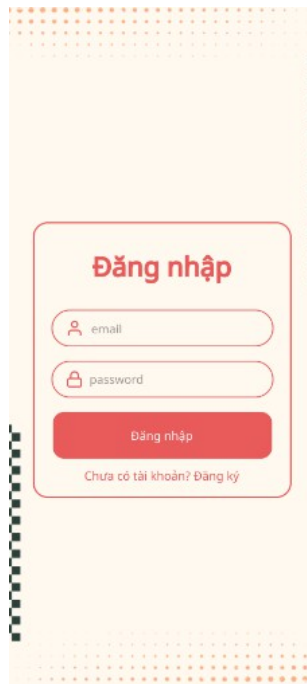
https://www.figma.com/proto/PdvarzZcv8zA5vPRRPdBaT/PM_DatDoAn?node-id=0-1&t=bf0DFVNCVScsOEOp-1

Ảnh kết xuất Figma:

3.1.1. Màn hình đăng ký



3.1.2. Màn hình đăng nhập



The image shows a mobile application login screen. It features a light orange background with a decorative pattern of small orange dots at the top and bottom. A central white rounded rectangle contains the login form. The form has a title 'Đăng nhập' in red, followed by two input fields: 'email' with a person icon and 'password' with a lock icon. Below these is a red 'Đăng nhập' button. At the bottom of the form, there is a link 'Chưa có tài khoản? Đăng ký'. The entire screen is framed by a black and white checkered border on the left and right sides.

Đăng nhập

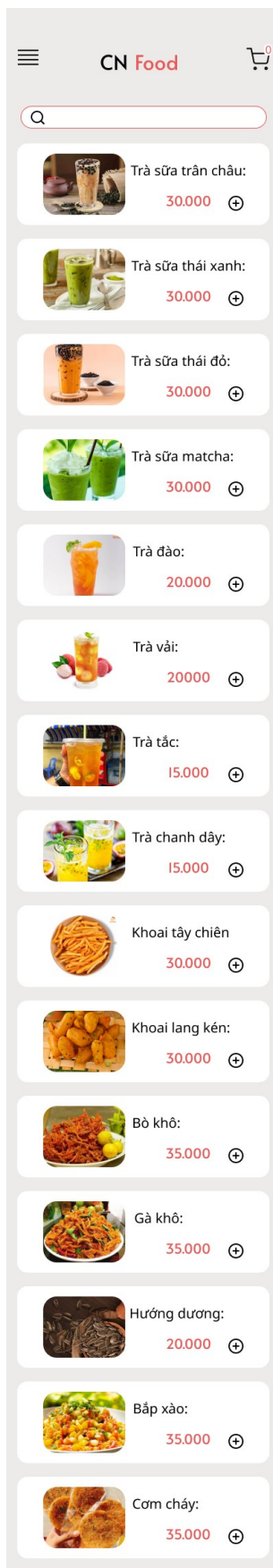
email

password

Đăng nhập

[Chưa có tài khoản? Đăng ký](#)

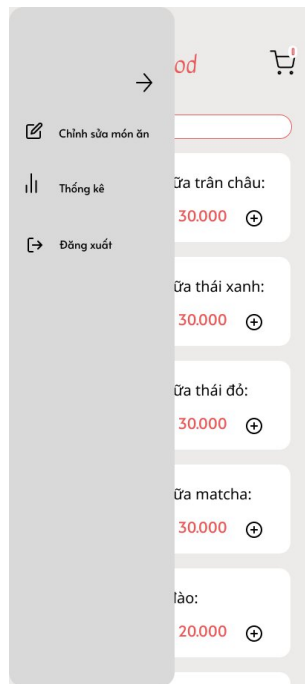
3.1.3. Màn hình trang chính



3.1.4. Màn hình giỏ hàng



3.1.5. Màn hình Navigation



3.1.6. Màn hình chỉnh sửa món ăn

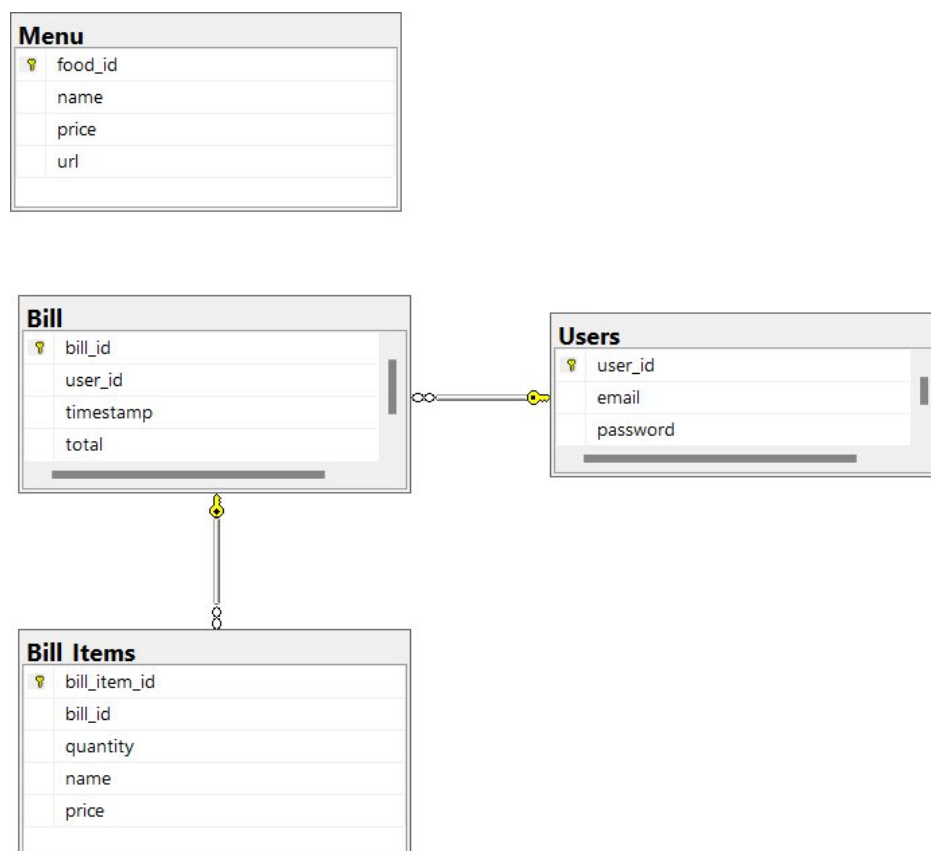


3.1.7. Màn hình thống kê



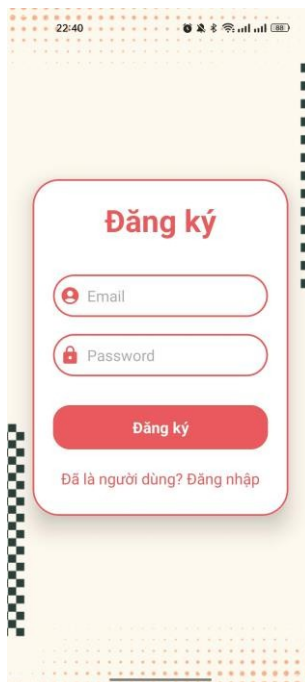
3.2 Thiết kế CSDL

<<Lược đồ>>



3.2. Giao diện ứng dụng

3.2.1. Màn hình đăng ký



22:40

Đăng ký

Email

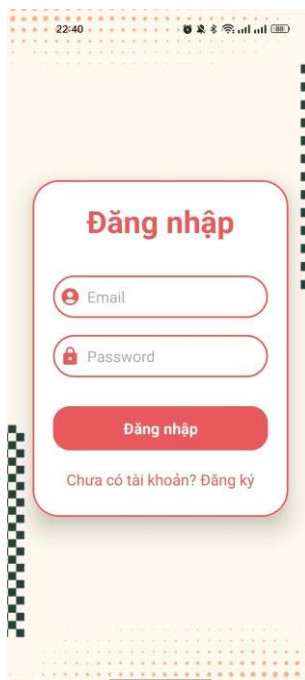
Password

Đăng ký

Đã là người dùng? Đăng nhập

The registration screen features a light yellow background with a subtle orange dot pattern at the top and bottom. A central white card with a red border contains the title 'Đăng ký' in red. Below the title are two input fields: 'Email' with an envelope icon and 'Password' with a lock icon. A red 'Đăng ký' button is positioned below the fields. At the bottom of the card, a link 'Đã là người dùng? Đăng nhập' is displayed. The screen is framed by a black and white checkered border on the right and bottom.

3.2.2. Màn hình đăng nhập



22:40

Đăng nhập

Email

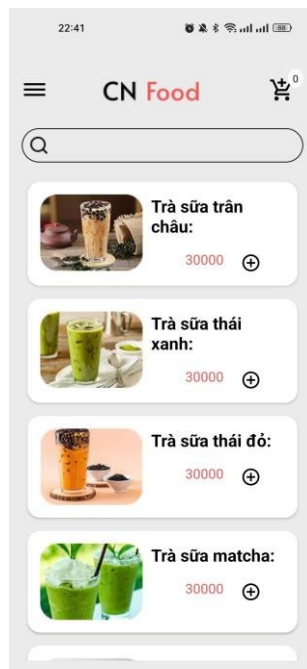
Password

Đăng nhập

Chưa có tài khoản? Đăng ký

The login screen has a similar design to the registration screen. It features a light yellow background with orange dot patterns. A central white card with a red border contains the title 'Đăng nhập' in red. Below the title are two input fields: 'Email' with an envelope icon and 'Password' with a lock icon. A red 'Đăng nhập' button is positioned below the fields. At the bottom of the card, a link 'Chưa có tài khoản? Đăng ký' is displayed. The screen is framed by a black and white checkered border on the right and bottom.

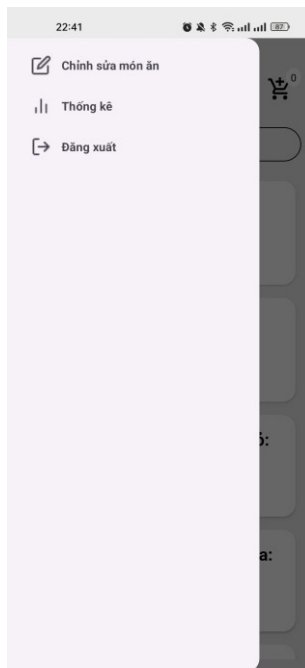
3.2.3. Màn hình trang chính



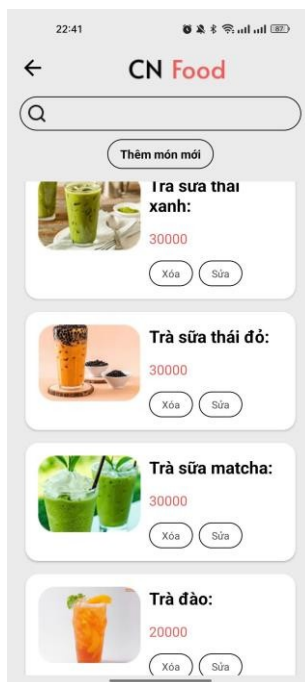
3.2.4. Màn hình giỏ hàng



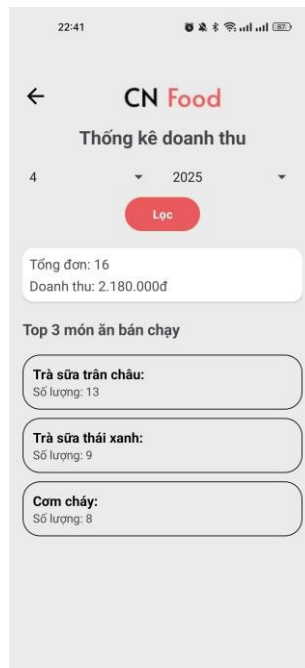
3.2.5. Màn hình Navigation



3.2.6. Màn hình chỉnh sửa món ăn



3.2.7. Màn hình thống kê



3.4 Code minh họa các chức năng cốt lõi

Đăng ký người dùng với Firebase Authentication

```
dk_button.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        String user=dk_email.getText().toString().trim();
        String pass=dk_password.getText().toString().trim();

        if(user.isEmpty()){
            dk_email.setError("Email không được bỏ trống");
        }
        if(pass.isEmpty()){
            dk_password.setError("Password không được bỏ trống");
        }else{
            auth.createUserWithEmailAndPassword(user,pass).addOnCompleteListener(new OnCompleteListener<AuthResult>() {
                @Override
                public void onComplete(@NonNull Task<AuthResult> task) {
                    if(task.isSuccessful()){
                        Toast.makeText( SignUpActivity.this, text: "Đăng ký thành công", Toast.LENGTH_SHORT).show();
                        startActivity(new Intent( packageContext: SignUpActivity.this, LoginActivity.class));
                    }else{
                        Toast.makeText( context: SignUpActivity.this, text: "Đăng ký không thành công" + task.getException().getMessage(), Toast.LEN
                    }
                }
            });
        }
    }
});
```

Tải danh sách món ăn từ Firebase và cập nhật lên RecyclerView

```
private void loadFoodData() {
    FirebaseDatabase database = FirebaseDatabase.getInstance();
    DatabaseReference menuRef = database.getReference( path: "menu");

    menuRef.get().addOnSuccessListener(dataSnapshot -> {
        if (dataSnapshot.exists()) {
            foodList.clear();
            for (DataSnapshot foodSnapshot : dataSnapshot.getChildren()) {
                String id=foodSnapshot.getKey();
                String name = foodSnapshot.child( path: "name").getValue(String.class);
                int price = foodSnapshot.child( path: "price").getValue(Integer.class);
                String img = foodSnapshot.child( path: "url").getValue(String.class);
                Food food = new Food(id,name, price, img);
                foodList.add(food);
            }
            foodAdapter.notifyDataSetChanged(); // Cập nhật dữ liệu vào adapter
        }
    }).addOnFailureListener(e -> {
        // Xử lý khi có lỗi
        Toast.makeText( context: MainActivity.this, text: "Lỗi khi tải dữ liệu", Toast.LENGTH_SHORT).show();
    });
}
```

Thêm món vào giỏ hàng

```
foodAdapter = new FoodAdapter(foodList, new FoodAdapter.OnAddFoodClickListener() {
    1 usage
    @Override
    public void onAddFoodClick(Food food) {
        orderCount++;
        number_order.setText(String.valueOf(orderCount));

        boolean found= false;

        for(CartItem item :cartList){
            if(item.getFood().getId().equals(food.getId())){
                item.setQuantity((item.getQuantity()+1));
                found=true;
                break;
            }
        }
        if(!found){
            cartList.add(new CartItem(food, quantity: 1));
        }
    }
});
```

Tìm kiếm món ăn theo tên

```
private void filterList(String Key){
    List<Food> filterList =new ArrayList<>();
    for(Food food : foodList){
        if(food.getNamefood().toLowerCase().contains(Key.toLowerCase())){
            filterList.add(food);
        }
    }
    foodAdapter.updateList(filterList);
}
```

Thống kê doanh thu theo tháng

```
for (DataSnapshot billSnapshot : snapshot.getChildren()) {
    String timestampStr = billSnapshot.child( path: "timestamp").getValue(String.class);
    Long total = billSnapshot.child( path: "total").getValue(Long.class);
    if (timestampStr == null || total == null) continue;
    try {
        SimpleDateFormat sdf = new SimpleDateFormat( pattern: "dd/MM/yyyy HH:mm:ss");
        Date date = sdf.parse(timestampStr);
        Calendar cal = Calendar.getInstance();
        cal.setTime(date);
        int billMonth = cal.get(Calendar.MONTH) + 1;
        int billYear = cal.get(Calendar.YEAR);
        if (billMonth == month && billYear == year) {
            totalOrders++;
            totalRevenue += total;
            for (DataSnapshot itemSnapshot : billSnapshot.child( path: "items").getChildren())
                String name = itemSnapshot.child( path: "name").getValue(String.class);
                Long quantity = itemSnapshot.child( path: "quantity").getValue(Long.class);
                if (name != null && quantity != null) {
                    name = name.trim();
                    int count = foodCountMap.getOrDefault(name, defaultValue: 0);
                    foodCountMap.put(name, count + quantity.intValue());
                }
            }
        }
    }
```


Chức năng xác nhận thanh toán

```
private void showConfirmDialog(Food food){
    Dialog dialog = new Dialog( context: this);
    dialog.setContentView(R.layout.dialog_confirm_pay);
    dialog.getWindow().setBackgroundDrawableResource(android.R.color.transparent);

    ImageView btnCancel = dialog.findViewById(R.id.btn_cancel);
    ImageView btnConfirm = dialog.findViewById(R.id.btn_confirm);

    btnCancel.setOnClickListener(v -> dialog.dismiss());

    btnConfirm.setOnClickListener(v -> {
        saveOrderToFirebase(); // Lưu đơn hàng
        cartItemList.clear(); // Xóa giỏ hàng
        cartAdapter.notifyDataSetChanged(); // Làm mới RecyclerView
        updateTotalPrice(); // Cập nhật lại tổng tiền
        dialog.dismiss();
        Toast.makeText( context: CardActivity.this, text: "Thanh toán thành công!", Toast.LENGTH_SHORT).show();
        startActivity(new Intent( packageContext: CardActivity.this, MainActivity.class));
    });

    dialog.show();
}
```

Lưu đơn hàng lên Firebase

```
private void saveOrderToFirebase() {
    DatabaseReference ordersRef = FirebaseDatabase.getInstance().getReference( path: "bills");
    String orderId = ordersRef.push().getKey(); // Tạo ID tự động
    Map<String, Object> orderData = new HashMap<>();
    List<Map<String, Object>> items = new ArrayList<>();
    int total = 0;
    for (CartItem item : cartItemList) {
        Map<String, Object> foodMap = new HashMap<>();
        foodMap.put("name", item.getFood().getNamefood());
        foodMap.put("price", item.getFood()..getPricefood());
        foodMap.put("quantity", item.getQuantity());
        items.add(foodMap);
        total += item.getFood().getPricefood() * item.getQuantity();
    }
    orderData.put("items", items);
    orderData.put("total", total);
    SimpleDateFormat sdf = new SimpleDateFormat( pattern: "dd/MM/yyyy HH:mm:ss");
    String currentTime = sdf.format(new Date());
    orderData.put("timestamp", currentTime);
    ordersRef.child(orderId).setValue(orderData);
}
```

KẾT LUẬN

1. Kết quả đạt được

Sau quá trình triển khai và phát triển, nhóm đã hoàn thành một ứng dụng gọi món trên nền tảng Android với các chức năng chính:

- Đăng ký, đăng nhập tài khoản thông qua Firebase Authentication.
- Hiện thị danh sách món ăn từ Firebase Realtime Database.
- Tìm kiếm món ăn theo tên trong thời gian thực.
- Thêm món ăn vào giỏ hàng, hiển thị và xác nhận đơn hàng.
- Navigation Drawer để truy cập các chức năng: chỉnh sửa món ăn, thống kê doanh thu, đăng xuất.
- Chức năng chỉnh sửa món ăn (tìm kiếm món theo tên, xóa món).
- Thống kê doanh thu theo tháng và top 3 món ăn bán chạy nhất.

2. Nhược điểm

- Ứng dụng chưa hỗ trợ thanh toán trực tuyến.
- Chưa có chức năng phân quyền người dùng (admin/khách hàng).
- Chưa hoàn thành xong thêm món mới vào menu, sửa món.
- Giao diện còn đơn giản, thiếu hiệu ứng trực quan và nâng cao trải nghiệm người dùng.
- Chưa có chức năng theo dõi đơn hàng theo trạng thái.

3. Hướng phát triển

- Bổ sung chức năng phân quyền người dùng để bảo mật và phân chia chức năng rõ ràng hơn.
- Hỗ trợ thanh toán điện tử qua ví điện tử như Momo, ZaloPay, VNPAY,...
- Cải tiến giao diện người dùng, thêm hiệu ứng chuyển cảnh và trải nghiệm mượt mà hơn.
- Phát triển tính năng quản lý đơn hàng nâng cao như trạng thái giao hàng, lịch sử đặt hàng.

TÀI LIỆU THAM KHẢO

- [1] David Flanagan, JavaScript: The Definitive Guide, 7th Edition, O'Reilly Media, 2020.
- [2] Adam Freeman, “Pro jQuery”, Apress, 2018.
- [3] Benjamin Jakobus, “Mastering Bootstrap 5”, Packt Publishing, 2018.
- [4] Youtube: [\(46\) Android Knowledge - YouTube](#)
- [5] ChatGpt: <https://chatgpt.com>